

Apache CloudStack 4.0.2

クイックインストールガイド

Apache CloudStack 簡易構築ガイド

エディション 0



CloudStack Apache [FAMILY Given]

法律上の通知

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache CloudStack is an effort undergoing incubation at The Apache Software Foundation (ASF).

Incubation is required of all newly accepted projects until a further review indicates that the infrastructure, communications, and decision making process have stabilized in a manner consistent with other successful ASF projects. While incubation status is not necessarily a reflection of the completeness or stability of the code, it does indicate that the project has yet to be fully endorsed by the ASF.

CloudStack® is a registered trademark of the Apache Software Foundation.

Apache CloudStack, the CloudStack word design, the Apache CloudStack word design, and the cloud monkey logo are trademarks of the Apache Software Foundation.

概要

このガイドでは、Apache CloudStackの初期導入で失敗しないことを重視しています。このドキュメントに載っている部品は誰でもアクセスできます。Apache CloudStackを使って、もっと複雑な構成を組むこともできますが、それはこのドキュメントのスコープ外です。

はじめに

1. 表記方法

- 1.1. 印刷における表記方法
- 1.2. 引用における表記方法
- 1.3. 注記および警告

2. Submitting Feedback and Getting Help

1. 概要

- 1.1. 実際私たちは何を構築するのでしょうか？
- 1.2. プロセスのハイレベルの概要

- 1.3. 要件
- 2. 環境
 - 2.1. オペレーティングシステム
 - 2.1.1. ネットワークの設定
 - 2.1.2. ホスト名
 - 2.1.3. SELinux
 - 2.1.4. NTP
 - 2.1.5. CCloudStack パッケージリポジトリの設定
 - 2.2. NFS
- 3. 管理サーバーのインストール
 - 3.1. データベースのインストールと設定
 - 3.2. インストール
 - 3.3. システムテンプレートのセットアップ
- 4. KVMのセットアップとインストール
 - 4.1. 前提条件
 - 4.2. インストール
 - 4.2.1. KVMの設定
- 5. 設定
 - 5.1. UIアクセス
 - 5.2. ゾーンのセットアップ
 - 5.3. ポッドの設定
 - 5.4. クラスタ
 - 5.4.1. プライマリストレージ
 - 5.4.2. セカンダリストレージ
- A. 変更履歴

はじめに

1. 表記方法

本ガイドは特定の単語や語句を強調したり、記載内容の特定部分に注意を引かせる目的で次のような表記方法を使用しています。

PDF版および印刷版では、[Liberation Fonts](#) セットから採用した書体を使用しています。ご使用のシステムに Liberation Fonts セットがインストールされている場合、HTML 版でもこのセットが使用されます。インストールされていない場合は代替として同等の書体が表示されます。注記: Red Hat Enterprise Linux 5 およびそれ以降のバージョンにはデフォルトで Liberation Fonts セットが収納されます。

1.1. 印刷における表記方法

特定の単語や語句に注意を引く目的で 4 種類の表記方法を使用しています。その表記方法および適用される状況は以下の通りです。

等幅の太字

シェルコマンド、ファイル名、パスなどシステムへの入力を強調するために使用しています。またキー配列やキーの組み合わせを強調するのにも使用しています。例えば、

```
現在作業中のディレクトリ内のファイル my_next_bestselling_novel の内容を表示させるには、シェルプロンプトで cat my_next_bestselling_novel コマンドを入力してから Enter を押してそのコマンドを実行します。
```

上記にはファイル名、シェルコマンド、キーが含まれています。すべて等幅の太字で表されているため文中内で見分けやすくなっています。

キーが 1 つの場合と複数のキーの組み合わせになる場合を区別するため、その組み合わせを構成するキー同士をハイフンでつないでいます。例えば、

```
Enter を押してコマンドを実行します。
```

```
1 番目の仮想ターミナルに切り替えるは、Ctrl+Alt+F2 を押します。X-Windows セッションに戻るには、Ctrl+Alt+F1 を押します。
```

最初の段落では押すべき 1 つのキーを特定して強調しています。次の段落では同時に押すべき 3 つのキーの組み合わせが 2 種類ありそれぞれ強調されています。

ソースコードの説明では 1 段落内で提示されるクラス名、メソッド、関数、変数名、戻り値を上記のように **等幅の太字** で表示します。例えば、

ファイル関連のクラス群はファイルシステムに対しては `filesystem`、ファイルには `file`、ディレクトリには `dir` をそれぞれ含みます。各クラスは個別に関連する権限セットを持っています。

プロポーショナルの太字

アプリケーション名、ダイアログボックスのテキスト、ラベル付きボタン、チェックボックスとラジオボタンのラベル、メニュータイトルとサブメニュータイトルなどシステム上で見られる単語や語句を表します。例えば、

メインメニューバーから `システム > 個人設定 > マウス` の順で選択し `マウスの個人設定` を起動します。`ボタン` タブ内で `左ききのマウス` チェックボックスをクリックしてから `閉じる` をクリックしマウスの主要ボタンを左から右に切り替えます (マウスを左ききの人が使用するのに適した設定にする)。

`gedit` ファイルに特殊な文字を挿入する場合は、メインメニューバーから `アプリケーション > アクセサリ > 文字マップ` の順で選択します。次に `文字マップ` メニューバーから `検索 > 検索...` と選択して `検索` フィールド内にその文字名を入力し `次` をクリックします。探している文字が `文字表` 内で強調表示されます。この強調表示された文字をダブルクリックすると `コピーするテキスト` フィールド内に置かれるので次に `コピー` ボタンをクリックします。ここでドキュメントに戻り `gedit` メニューバーから `編集 > 貼り付け` を選択します。

上記には、アプリケーション名、システム全体のメニュー名と項目、アプリケーション固有のメニュー名、GUI インタフェースで見られるボタンやテキストがあります。すべてプロポーショナルの太字で表示されているため文中内で見分けやすくなっています。

等幅の太字で且つ斜体 または プロポーショナルの太字で且つ斜体

等幅の太字やプロポーショナルの太字はいずれであっても斜体の場合は置換可能なテキストか変化するテキストを示します。斜体は記載されている通りには入力しないテキスト、あるいは状況に応じて変化する出力テキストを表します。例えば、

`ssh` を使用してリモートマシンに接続するには、シェルプロンプトで `ssh username@domain.name` と入力します。リモートマシンが `example.com` であり、そのマシンで使用しているユーザー名が `john` なら `ssh john@example.com` と入力します。

`mount -o remount file-system` コマンドは指定したファイルシステムを再マウントします。例えば、`/home` ファイルシステムを再マウントするコマンドは `mount -o remount /home` になります。

現在インストールされているパッケージのバージョンを表示するには、`rpm -q package` コマンドを使用します。結果として次を返してきます、`package-version-release`。

上記の太字斜体の単語 — `username`、`domain.name`、`file-system`、`package`、`version`、`release` に注目してください。いずれもコマンドを発行するときに入力するテキスト用のプレースホルダーかシステムにより出力されるテキスト用のプレースホルダーになっています。

タイトル表示のような標準的な使用の他、斜体は新しい重要な用語が初めて出現する場合にも使用されます。例えば、

`Publican` は `DocBook` の発行システムです。

1.2. 引用における表記方法

端末の出力とソースコード一覧は、視覚的に周囲の文から区別されています。

端末に送信される出力は `mono-spaced roman` (等幅の Roman) にセットされるので以下のように表示されます。

```
books      Desktop  documentation  drafts  mss      photos  stuff  svn
books_tests Desktop1  downloads      images  notes    scripts svgs
```

ソースコードの一覧も `mono-spaced roman` (等幅の Roman) でセットされますが、以下のように強調表示されます。

```
package org.jboss.book.jca.ex1;
import javax.naming.InitialContext;

public class ExClient
{
    public static void main(String args[])
        throws Exception
    {
        InitialContext iniCtx = new InitialContext();
        Object ref = iniCtx.lookup("EchoBean");
        EchoHome home = (EchoHome) ref;
        Echo echo = home.create();

        System.out.println("Created Echo");

        System.out.println("Echo.echo('Hello') = " + echo.echo("Hello"));
    }
}
```

1.3. 注記および警告

情報が見逃されないう 3 種類の視覚的なスタイルを使用して注意を引いています。

注記

注記は説明している部分に対するヒントや近道あるいは代替となる手段などになります。注記を無視しても悪影響はありませんが知っておくと便利なコツを見逃すことになるかもしれません。



重要

重要ボックスは見逃しやすい事項を詳細に説明しています。現在のセッションにのみ適用される設定上の変更点、更新を適用する前に再起動が必要なサービスなどがあります。重要ボックスを無視してもデータを喪失するような結果にはなりませんがいらいら感やフラストレーションが生じる可能性があります。



警告

警告は無視しないでください。警告を無視するとデータを喪失する可能性が非常に高くなります。

2. Submitting Feedback and Getting Help

If you find a typographical error in this manual, or if you have thought of a way to make this manual better, we would love to hear from you! Please submit a bug: <https://issues.apache.org/jira/browse/CLOUDSTACK> against the component **Doc**.

If you have a suggestion for improving the documentation, try to be as specific as possible when describing it. If you have found an error, please include the section number and some of the surrounding text so we can find it easily.

Better yet, feel free to submit a patch if you would like to enhance the documentation. Our documentation is, along with the rest of the Apache CloudStack source code, kept in the project's git repository.

The most efficient way to get help with Apache CloudStack is to ask on the mailing lists.

The Apache CloudStack project has mailing lists for users and developers. These are the official channels of communication for the project and are the best way to get answers about using and contributing to CloudStack. It's a good idea to subscribe to the users@cloudstack.apache.org mailing list if you've deployed or are deploying CloudStack into production, and even for test deployments.

The CloudStack developer's mailing list (dev@cloudstack.apache.org) is for discussions about CloudStack development, and is the best list for discussing possible bugs in CloudStack. Anyone contributing to CloudStack should be on this mailing list.

To post to the lists, you'll need to be subscribed. See the [CloudStack Web site](#) for instructions.

第1章 概要

1.1. 実際私たちは何を構築するのでしょうか？

1.2. プロセスのハイレベルの概要

1.3. 要件

Infrastructure-as-a-Service (IaaS) のクラウドは、複雑な構築物になり得ます。また、定義上過剰な選択肢があるので、それがまた、経験のある管理者であったとしても、クラウドプラットフォームを構築する初心者には混乱を招きがちです。この runbook の目的は、単純な手順を提供し、CloudStack をできるだけトラブルなく動かせるようにすることです。

1.1. 実際私たちは何を構築するのでしょうか？

この runbook では、KVM, CnetOS 6.3, NFS ストレージ、フラットな Layer-2 ネットワーク上で layer-3 ネットワーク隔離（セキュリティグループ）、そしてそれら全てをひとつのマシン上で実行するという構成に焦点を絞っています。

KVM (Kernel-based Virtual Machine) はLinuxカーネルの仮想化技術です。KVMは、ハードウェア仮想化機能を持つプロセッサ上で動作し、ネイティブの仮想化をサポートします。

セキュリティグループは分散されたファイアウォールとして動き、仮想マシンのグループへのアクセスをコントロールします。

1.2. プロセスのハイレベルの概要

CloudStackのインストールを実際に始める前に、ベースとなるオペレーティングシステムのインストールから始めます。そして、それをいくつかのタイプのストレージとして使う NFS サーバーとして構成します。管理サーバーをインストールして、システム VM をダウンロードして、最後にエージェントソフトウェアをインストールします。そこまでできたら、クラウド全体を設定するために、CloudStack のウェブインターフェースで多くの時間を過ごすことになります。

1.3. 要件

この runbook を完了するために、次のものがが必要です：

1. 少なくとも1台のハードウェア仮想化をサポートするコンピューター
2. [CentOS 6.3 x86_64 最小インストール CD](#)
3. xxx.xxx.xxx.1 で始まるゲートウェイを持つ /24 ネットワークで、DHCP が存在せず、かつ、CloudStack を実行しているコンピューターでは動的アドレスを使っていないこと。念のために言うと、簡単ためこの条件にしています。

第2章 環境

2.1. オペレーティングシステム

- 2.1.1. ネットワークの設定
- 2.1.2. ホスト名
- 2.1.3. SELinux
- 2.1.4. NTP
- 2.1.5. CCloudStack パッケージリポジトリの設定

2.2. NFS

始める前に、CloudStackをインストールするための環境を準備する必要があります。ここから準備のためのステップを始めます。

2.1. オペレーティングシステム

CentOS 6.3 x86_64の最小インストールISOを使って、マシンにCentOSをインストールしましょう。このインストールでは、ほぼデフォルトの設定で大丈夫です。

インストールが完了したら、新しくインストールしたマシンにSSH経由で、rootユーザーとして接続したくなるでしょう。注意しないといけないのは、本番環境ではrootログインは許可すべきでないということです。ですので、インストールと設定が済んだ後は、リモートログインをオフにするようにしましょう。

2.1.1. ネットワークの設定

デフォルトではネットワークはハードウェアに合った設定になっていないので、作成する環境で動くように設定する必要があります。ここでは、環境内にDHCPサーバーが無いことを前提にしているので、NICを手動で設定する必要があります。また、これは検証目的ということもあるので、eth0が唯一かつ使用するNICであることを前提とします。

コンソール経由で接続し、rootとしてログインします。`/etc/sysconfig/network-scripts/ifcfg-eth0`ファイルをチェックします。これはデフォルトではこのようになっているはずで：

```
DEVICE="eth0"
HWADDR="52:54:00:B9:A6:C0"
NM_CONTROLLED="yes"
ONBOOT="no"
```

残念ながら、この設定はネットワークに接続できる設定になっていませんし、CloudStackで使用するという目的にも適していません。このファイルを編集して、IPアドレスやネットマスクなどを以下の例のように修正したいと思います：



ハードウェアアドレス

ハードウェアアドレス (MACアドレス) はこの例では使用しません。これはNICに固有の設定なので、HWADDRの項目にすでに入っている値をそのまま使うべきです。

```
DEVICE=eth0
HWADDR=52:54:00:B9:A6:C0
NM_CONTROLLED=no
ONBOOT=yes
BOOTPROTO=none
IPADDR=172.16.10.2
NETMASK=255.255.255.0
GATEWAY=172.16.10.1
DNS1=8.8.8.8
DNS2=8.8.4.4
```



IPアドレス

このドキュメント全体を通じて、作成しようとしているCloudStackの環境が/24のネットワークを持っていると仮定します。RFC 1918のネットワークであれば何でも構わないのですが、マシンアドレスがここで設定するものとマッチしているとみなして進めます。ということで、ここでは**172.16.10.2**を設定しますが、もちろん、192.168.55.0/24 ネットワークを使うのであれば、**192.168.55.2**を設定しても構いません。

ここまでで、設定ファイルは正しくセットアップできたので、いくつかのコマンドを実行し、ネットワークを開始する必要があります。

```
# chkconfig network on
```

```
# service network start
```

2.1.2. ホスト名

CloudStackではホスト名が正しく設定されている必要があります。もし、インストール時にデフォルトを選択している場

場合は、ホスト名は localhost.localdomain になっています。これをテストするためにこれを実行します：

```
# hostname --fqdn
```

ここでは、これが返ってくるはずですが：

```
localhost
```

この状態を修正するために、`/etc/hosts` ファイルを修正してホスト名を設定します。この例のように修正してください：

```
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
172.16.10.2 srvr1.cloud.priv
```

ファイルを修正したら、このコマンドでネットワークを再起動します：

```
# service network restart
```

ここで、`hostname --fqdn` コマンドを用いて再チェックし、FQDNが返ってくることを確認します。

2.1.3. SELinux

CloudStackが正しく動作するために、SELinuxはpermissiveに設定されている必要があります。ここでは今後の起動時の設定と、現在動作しているシステムの設定の両方を変更します。

動作中のシステムのSELinuxをpermissiveに設定するために、以下のコマンドを実行します：

```
# setenforce 0
```

この状態を保つために、`/etc/selinux/config` ファイルを以下に示す例のように設定し、permissiveの状態を反映させます。

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
SELINUX=permissive
# SELINUXTYPE= can take one of these two values:
#   targeted - Targeted processes are protected,
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

2.1.4. NTP

NTPの設定は、クラウド内の全てのサーバーの時刻を同期させるために必要です。しかし、NTPはデフォルトではインストールされていません。なので、ここでNTPをインストールして設定します。インストールは以下のように行います：

```
# yum -y install ntp
```

ここでの目的からすると、実際のデフォルトの設定で十分です。なので、単に有効化して起動時に開始するように設定すれば良いです。以下のように：

```
# chkconfig ntpd on
```

```
# service ntpd start
```

2.1.5. CCloudStack パッケージリポジトリの設定

CloudStack パッケージリポジトリを使うようにマシンを設定します。



次のリポジトリは Apache CloudStack プロジェクトの公式なリポジトリではありません。

Apache CloudStack の公式なリリースはソースコードです。ですから、いわゆる「公式な」バイナリはありません。完全なインストールガイドには、ソースコードリリースを取得して、そこからRPMを生成しyumリポジトリを作る方法が記載されています。このガイドでは簡単のため、コミュニティで提供されているyumリポジトリのひとつを使います。

CloudStack リポジトリを追加するには、`/etc/yum.repos.d/cloudstack.repo` を作成し、次の内容を書き込みます。

```
[cloudstack]
name=cloudstack
baseurl=http://cloudstack.appt-get.eu/rhel/4.0/
enabled=1
gpgcheck=0
```

2.2. NFS

この環境では、NFSをプライマリとセカンダリーストレージの両方で使用します。この用途のために、2つのNFS共有を設定します。まず、`nfs-utils`をインストールするところから始めます。

```
# yum install nfs-utils
```

二つの異なる共有を提供するようにNFSを設定します。これは、比較的簡単な `/etc/exports` ファイルによって制御されています。以下のような内容になっていることを確認します：

```
/secondary *(rw,async,no_root_squash)
/primary *(rw,async,no_root_squash)
```

気づいたかもしれませんが、ここで定義されている2つのディレクトリは(まだ)システムに存在しません。以下のコマンドで、これらのディレクトリを作成し、パーミッションを正しく設定します：

```
# mkdir /primary
# mkdir /secondary
```

CentOS 6.xリリースではNFSv4をデフォルトで使用します。NFSv4ではドメインの設定が全てのクライアントで一貫している必要があります。今回の場合は、ドメインは `cloud.priv` ですので、`/etc/idmapd.conf` のドメインの設定がアンコメントされており、以下のように設定されていることを確認します。

```
Domain = cloud.priv
```

`/etc/sysconfig/nfs` ファイルの設定値をアンコメントします。

```
LOCKD_TCPPORT=32803
LOCKD_UDPPORT=32769
MOUNTD_PORT=892
RQUOTAD_PORT=875
STATD_PORT=662
STATD_OUTGOING_PORT=2020
```

ファイアーウォールを設定し、NFSの受信接続を許可します。`/etc/sysconfig/iptables` ファイルを編集します。

```
-A INPUT -s 172.16.10.0/24 -m state --state NEW -p udp --dport 111 -j ACCEPT
-A INPUT -s 172.16.10.0/24 -m state --state NEW -p tcp --dport 111 -j ACCEPT
-A INPUT -s 172.16.10.0/24 -m state --state NEW -p tcp --dport 2049 -j ACCEPT
-A INPUT -s 172.16.10.0/24 -m state --state NEW -p tcp --dport 32803 -j ACCEPT
-A INPUT -s 172.16.10.0/24 -m state --state NEW -p udp --dport 32769 -j ACCEPT
-A INPUT -s 172.16.10.0/24 -m state --state NEW -p tcp --dport 892 -j ACCEPT
-A INPUT -s 172.16.10.0/24 -m state --state NEW -p udp --dport 892 -j ACCEPT
-A INPUT -s 172.16.10.0/24 -m state --state NEW -p tcp --dport 875 -j ACCEPT
-A INPUT -s 172.16.10.0/24 -m state --state NEW -p udp --dport 875 -j ACCEPT
-A INPUT -s 172.16.10.0/24 -m state --state NEW -p tcp --dport 662 -j ACCEPT
-A INPUT -s 172.16.10.0/24 -m state --state NEW -p udp --dport 662 -j ACCEPT
```

以下のコマンドで、iptablesのサービスを再起動させます：

```
# service iptables restart
```

NFSサービスを起動時に開始するように設定し、実際に今、以下のコマンドを実行してNFSサービスを開始します。

```
# service rpcbind start
# service nfs start
# chkconfig rpcbind on
# chkconfig nfs on
```

第3章 管理サーバーのインストール

3.1. データベースのインストールと設定

3.2. インストール

3.3. システムテンプレートのセットアップ

ここから、CloudStackの管理サーバーといくつかの関連するコンポーネントのインストールを開始します。

3.1. データベースのインストールと設定

まず、MySQLのインストールから始め、CloudStackが正しく動くようにいくつかのオプションを設定します。

MySQLをインストールするために、以下のコマンドを実行します：

```
# yum -y install mysql-server
```

MySQLのインストールが終了したら、`/etc/my.cnf` の設定をいくつか変更します。具体的には、`[mysqld]` セクションに以下の設定を追加する必要があります。

```
innodb_rollback_on_timeout=1
innodb_lock_wait_timeout=600
max_connections=350
log-bin=mysql-bin
binlog-format = 'ROW'
```

ここまででMySQLが正しく設定されたので、開始することができます。また、起動時にMySQLが開始されるように設定します。

```
# service mysqld start
# chkconfig mysqld on
```

3.2. インストール

マネジメントサーバをインストールします。次のコマンドを実行してください:

```
# yum -y install cloud-client
```

アプリケーション自体はインストールできたので、データベースをセットアップします。次のコマンドとオプションで実行します:

```
# cloud-setup-databases cloud:password@localhost --deploy-as=root
```

プロセスが終了すると、"CloudStack has successfully initialized the database."というようなメッセージが表示されます。

データベースが作成されたので、管理サーバーのセットアップの最後のステップに入ることができます。以下のコマンドを実行してください:

```
# cloud-setup-management
```

3.3. システムテンプレートのセットアップ

CloudStackはいくつものシステムVMを使って、仮想マシンのコンソールへのアクセスや様々なネットワークサービス、ストレージの管理などの機能を提供します。このステップで、これらのシステムイメージを取得して、クラウドを起動する際に展開するための準備を行います。

ここで、システムVMテンプレートをダウンロードし、いまマウントした共有に展開する必要があります。管理サーバーにはシステムVMイメージを正しく操作するためのスクリプトが含まれています。

```
# /usr/lib64/cloud/common/scripts/storage/secondary/cloud-install-sys-tmpl -m /secondary
-u http://download.cloud.com/templates/acton/acton-systemvm-02062012.qcow2 -h kvm -F
```

これで管理サーバーのセットアップは終わりです。まだ CloudStack の設定は残っていますが、ハイパーバイザーのセットアップの後で行います。

第4章 KVMのセットアップとインストール

4.1. 前提条件

4.2. インストール

4.2.1. KVMの設定

KVMがこれから使用するハイパーバイザーです。ここではすでにハイパーバイザーホストに対して行った初期設定を振り返り、エージェントソフトウェアのインストールを取り上げます。追加のKVMノードをCloudStack環境に追加する場合は同じ手順を使うことができます。

4.1. 前提条件

ここでは管理サーバーをコンピュータードとしても使用します。つまり、管理サーバーをセットアップした際に、すでに多くの必要な事前手順はすでに実行しているということです。しかし、明確にするために、ここにリストします。そのステップは:

1. [「ネットワークの設定」](#)
2. [「ホスト名」](#)
3. [「SELinux」](#)
4. [「NTP」](#)
5. [「CloudStack パッケージリポジトリの設定」](#)

もちろん、あなたは管理サーバー用にそれをする必要はありません。しかし、どんな追加のホストも上記の手順を終える必要があるでしょう。

4.2. インストール

KVM agentのインストール自体はコマンドひとつで終わる簡単なものですが、その後いくつか設定しなければいけません。

```
# yum -y install cloud-agent
```

4.2.1. KVMの設定

KVM 設定には libvirt と QEMU の二つのパートがあります。

4.2.1.1. QEMU 設定

KVM の設定は一つの項目しかないので、比較的シンプルです。ここでは QEMU の VNC の設定を変更する必要があります。/etc/libvirt/qemu.conf を編集して、以下の行が存在し、コメントアウトされていないことを確認します。

```
vnc_listen=0.0.0.0
```

4.2.1.2. Libvirt 設定

CloudStack は仮想マシンの管理に libvirt を使います。ですから libvirt が正しく設定されていなければなりません。Libvirt は cloud-agent の dependency に指定されているので、インストール済みになっているはずです。

1. Live migration ができるためには、libvirt は非セキュアな TCP 接続を開いていなければなりません。Multicast DNS 広告を試みるオプションも無効化する必要があります。両方とも /etc/libvirt/libvirtd.conf で設定できます。

次のパラメータを設定します:

```
listen_tls = 0
```

```
listen_tcp = 1
```

```
tcp_port = "16059"
```

```
auth_tcp = "none"
```

```
mdns_adv = 0
```

2. libvirtd.conf で "listen_tcp" を有効にするだけでなく、パラメータも変更しなければいけません:

RHEL や CentOS では /etc/sysconfig/libvirtd を修正します:

次の行のコメントアウトをはずします:

```
#LIBVIRT_ARGS="--listen"
```

Ubuntu では /etc/init/libvirt-bin.conf を修正します。

次の行を変更します (ファイルの最後):

```
exec /usr/sbin/libvirtd -d
```

(-l を追加)

```
exec /usr/sbin/libvirtd -d -l
```

3. libvirt の再起動

RHEL や CentOS では:

```
$ service libvirtd restart
```

Ubuntu では:

```
$ service libvirt-bin restart
```

4.2.1.3. KVM 設定完了

これで KVM のインストールと設定が終了です。ここで、CloudStack の UI に戻り、クラウド環境の実際の設定に入っていきます。

第5章 設定

5.1. UI アクセス

5.2. ゾーンの設定

5.3. ポッドの設定

5.4. クラスタ

5.4.1. プライマリストレージ

5.4.2. セカンダリストレージ

前に記述したように、デフォルトではセキュリティグループを使用して分離を提供します。これは、フラットな Layer-2 ネットワークを使用することを意味します。また、セットアップを単純化することで、クイックインストーラーを使用できるということも意味します。

5.1. UI アクセス

CloudStack の Web インターフェースにアクセスするためには、単にブラウザに `http://172.16.10.2:8080/client` を指定すれ

ばよいです。デフォルトのユーザー名は'admin'で、デフォルトのパスワードは'password'です。スプラッシュスクリーンでは、CloudStackのセットアップを行う、いくつかのオプションを選択することができます。ここでは、**基本インストールを実行する** オプションを選びます。

ここで管理者ユーザーのパスワードを変更するように要求するプロンプトが表示されますので、変更してください。

5.2. ゾーンのセットアップ

ゾーンはCloudStackにおける最大の組織単位で、1つ作成していきます。現在この画面が表示されているはずですが、5つの情報を入力する必要があります。

1. 名前 - ここでは作成するクラウドに"Zone1"と設定します。
2. 外部DNS 1 - ここでは作成するクラウドに'8.8.8.8'と設定します。
3. 外部DNS 2 - ここでは作成するクラウドに'8.8.4.4'と設定します。
4. 内部DNS 1 - ここでは作成するクラウドに'8.8.8.8'と設定します。
5. 内部DNS 2 - ここでは作成するクラウドに'8.8.4.4'と設定します。

DNS設定の注意

CloudStackは内部と外部のDNSを区別しています。内部DNSは内部のみのホスト名、例えば、NFSサーバーのDNS名などを解決できることを想定しています。外部DNSはゲストVMがパブリックのIPアドレスを解決するために提供されます。両方のタイプに同じDNSサーバーを指定することも可能ですが、もしそうするならば、そのDNSサーバーは内部のIPもパブリックのIPも両方解決できる必要があります。この手順では、内部リソースには名前を使用せず、実際には同じ外部のリソースを見るように設定するので、DNSサーバーを追加することを要件には含めていません。

5.3. ポッドの設定

ゾーンを追加したので、次のステップでは、ポッドに関する情報を尋ねられます。それは4つのアイテムで構成されています。

1. 名前 - ここでは「Pod1」を使用します。
2. ゲートウェイ - 「172.16.10.1」をゲートウェイとして使用します。
3. ネットマスク - 「255.255.255.0」を使用します。
4. システムIPのレンジ - 「172.16.10.10-172.16.10.20」を使用します。
5. ゲストのゲートウェイ - 「172.16.10.1」を使用します。
6. ゲストのネットマスク - 「255.255.255.0」を使用します。
7. ゲストIPのレンジ - 「172.16.10.30-172.16.10.200」を使用します。

5.4. クラスタ

ゾーンを追加したので、後はクラスタについての設定をいくつか追加するだけです。

1. 名前 - 「Cluster1」を使用します。
2. ハイパーバイザー - 「KVM」を選びます。

ここではクラスタに最初のホストを追加するように指示されます。必要なのはほんのちょっとの情報だけです。

1. ホスト名 - DNSサーバーを設定してないので、「172.16.10.2」を設定します。
2. ユーザー名 - 'root'を使用します。
3. パスワード - rootユーザーのOSのパスワードを入力します。

5.4.1. プライマリストレージ

クラスタが設定できたので、プライマリストレージの情報が聞かれます。ストレージタイプとして「NFS」を選択し、フィールドに以下の値を入力します：

1. 名前 - 「Primary1」を使用します。
2. サーバー - 「172.16.10.2」を使用します。
3. パス - 「/primary」を指定します。

5.4.2. セカンダリストレージ

これが新しいゾーンの場合、セカンダリストレージの情報を求められます。以下のように入力します：

1. NFSサーバー - 「172.16.10.2」を使用します。
2. パス - /secondaryを使用します。

「起動」をクリックすると、クラウドのセットアップが開始されます。インターネット接続の早さによりますが、セットアップが完了するまでには数分かかります。

変更履歴

更新履歴
Apache CloudStack 4.0.1 へのアップデート

改訂 0-0
publican での初版

Mon Jun 25 2012